

Requirement

[Go up to Terms and Definitions](#)

Discussion

A Requirement is a statement of a necessary capability, behavior, quality, or constraint that an identified subject SHALL satisfy.

A Requirement can specify:

- A mission objective
- An operational capability
- A function
- A required behavior
- A prohibited behavior
- A security property
- A logging or auditing obligation
- A reliability property
- A performance constraint
- A maintainability property
- A data-management obligation
- An interoperability property
- Another necessary capability, behavior, quality, or constraint

A Requirement should identify:

- A stable Requirement Identifier
- The responsible subject
- The required capability, behavior, quality, or constraint
- The object to which the Requirement applies
- The applicable conditions
- The applicable limits
- The applicable source
- The applicable rationale
- The applicable verification method
- The applicable [Acceptance Criteria](#)
- The applicable delivery phase
- The implementation status
- The Requirement status
- The applicable issues
- Its [Provenance](#)
- Its [Traceability](#)

A Requirement can derive from:

- A stakeholder need
- A mission objective

- An operational need
- A law
- A regulation
- A contract
- A standard
- A [Policy](#)
- A [Governance Policy](#)
- An architectural decision
- A higher-level Requirement
- A risk
- A constraint
- Another authoritative source

Definition

statement of a necessary capability, behavior, quality, or constraint that an identified subject SHALL satisfy

Source

Adapted from:

- ISO/IEC/IEEE 29148, Systems and software engineering — Life cycle processes — Requirements engineering
- OMG Specification Discipline and Authoring (SDA)
- Common Specification Weakness Enumeration (CWE), Version 1.0
- DIDO Reference Architecture
- DIDO Reference Implementation Conceptual Model
- DIDO-TE draft Requirements Register

ISO/IEC/IEEE 29148 provides the systems and software engineering framework for requirements and requirements-engineering information.

OMG SDA establishes the authoring discipline for clear, precise, atomic, and independently interpretable normative statements.

The Common Specification Weakness Enumeration establishes rules and defect patterns for evaluating normative content, including ambiguity, non-testable requirements, weak language, implementation leakage, and multiple normative obligations.

Note

A Requirement differs from a [Policy](#):

- A Policy establishes direction, rules, or constraints

- A Requirement states a necessary capability, behavior, quality, or constraint that an identified subject SHALL satisfy

A Policy can provide the source or rationale for one or more Requirements.

A Requirement differs from [Acceptance Criteria](#):

- A Requirement states the obligation
- Acceptance Criteria establish the conditions used to determine whether the obligation has been satisfied

A Requirement differs from an objective:

- An objective identifies a desired outcome
- A Requirement states an obligation that an identified subject SHALL satisfy

An objective can provide the source for one or more Requirements.

A Requirement differs from a design decision:

- A Requirement specifies a necessary capability, behavior, quality, or constraint
- A design decision selects how an implementation satisfies one or more Requirements

A Requirement should remain independent of implementation technology unless the controlling source requires a particular technology.

A conforming Requirement must:

- Identify the responsible subject
- Identify the required capability, behavior, quality, or constraint
- Identify the object to which the Requirement applies
- Express a single, independently interpretable obligation
- Use defined terminology
- Use precise and unambiguous language
- Use the applicable normative modal verb
- State necessary conditions and limits explicitly
- Support objective verification
- Preserve Traceability to its source

A Requirement that combines independently verifiable obligations should be decomposed into separate derived Requirements.

When decomposition occurs:

- The parent Requirement retains its stable Requirement Identifier
- Each derived Requirement receives a stable subordinate identifier
- Each derived Requirement identifies the parent under **Derived From**
- The decomposition preserves the complete intent of the parent Requirement
- The derived Requirements remain independently verifiable

A Requirement Statement uses:

- SHALL to express an obligation

- SHALL NOT to express a prohibition
- MAY to express permission when the specification needs to state an allowed option

SHOULD expresses a recommendation and does not establish a mandatory Requirement.

A Requirement should not depend on vague or subjective expressions such as:

- Appropriate
- Adequate
- Efficient
- Flexible
- Rapid
- Robust
- Seamless
- Sufficient
- User-friendly
- As needed
- Where practical
- To the extent possible

When such a characteristic is necessary, the Requirement or its Acceptance Criteria must establish measurable and bounded conditions.

A Requirement Identifier remains stable after external citation. Changes to the Requirement Statement must preserve review history, Provenance, Traceability, and the relationship to applicable verification records.

Example

The following statement is not objectively verifiable:

DIDO-TE SHALL execute tests rapidly.

The term rapidly does not establish a measurable performance constraint.

The revised Requirement states:

*DIDO-TE SHALL complete the selected **Test Run** within the maximum execution duration established by the **Acceptance Criteria**.*

The Acceptance Criteria identify:

- The selected Test Run
- The activities included in the measurement
- The starting event
- The completion event
- The maximum permitted duration
- The measurement conditions

- The required measurement precision
- The permitted exclusions and tolerances
- The treatment of failed, incomplete, interrupted, or repeated Test Runs

This structure permits objective verification and preserves Traceability among the Requirement, Acceptance Criteria, Test Run, measured duration, and verification result.

© 2026 Dido Solutions, Inc. and Jackrabbit Consulting, Inc.

From:

<https://wiki.didosolutions.com/> - **Dido Solutions, Inc Wiki**

Permanent link:

https://wiki.didosolutions.com/dido/99_annexes/annex-b-terms-and-definitions/r/requirement

Last update: **2026/08/04 14:38**

